Automatic Problem Identification by Disease Category Using Deep Learning

Ching-Huei Tsou, PhD, Safa Messaoud, MS, Jennifer J. Liang, MD, Murthy V. Devarakonda, PhD
IBM Research, Yorktown Heights, NY, USA

1 Introduction

Physicians struggle to assimilate the vast amounts of data in Electronic Health Records (EHRs) and report decreased productivity and low satisfaction[1]. Problem-oriented summarization has the potential to reduce the cognitive load. Earlier we developed a manually feature-engineered machine learning method to automatically generate a problem list from a longitudinal patient record using natural language processing (NLP)[2]. While the overall results were good, further analysis showed that there is room for improvement, especially for certain disease categories. One obvious approach to improving the results is to build models that leverage features specific to disease categories, where ICD-9 top level is used for categorization. However, manually engineering features for each of the disease categories is both tedious and error prone. Recent developments in representation learning have shown promise in such tasks[3]. This research explores using deep learning to automate problem list generation, which represents an important clinical application area of NLP and learning methods.

2 Methods

We contextualized the problem list definition given by the Center for Medicare and Medicaid Services (CMS)[4] by considering the problem list in the setting of a comprehensive health assessment. We acquired 399 de-identified longitudinal patient records containing 33,815 clinical notes and associated comprehensive patient data from Cleveland Clinic under IRB approval. A gold standard was created for them through adjudicated manual annotations by two medical experts for each record. The general process of problem list generation is one of identifying non-negated disorders in clinical notes that map to SNOMED CT CORE subset as candidate problems, and further extracting features for each candidate problem and training a machine learning model to classify each candidate as a problem or a non-problem for the patient.

In the baseline, we manually engineered several features based on clinical, lexical, structural, temporal, and epidemiological aspects of the candidate problems extracted from the narratives in the clinical notes as well as from the structured parts of a patient record (approximately 100 multi-valued features). The features included:

- Context of the candidate problem mention in a clinical note, e.g., in Assessment and Plan (A&P) section of the note
- Prior problem incidence, in 990 Cleveland Clinic patient records and in the general population in the USA
- Patient being on one or more medications for the problem
- Existence of a formal diagnosis, e.g., coded in ICD-9, by a physician in the patient record for the problem
- Indication that the problem is a chronic or recurrent condition, e.g., through frequent mentions in A&P sections
- Confidence score of our NLP component in recognizing/normalizing the surface form of the problem

A single supervised model was trained using Alternating Decision Tree (ADT)[5] and tested on the gold standard for all disease categories.

2.1 Deep Learning Autoencoder Methods: One Model for All Categories and a Model for Each Category

To improve upon the baseline through per disease-category based models, we used several variations of autoencoders[6,7] to discover customized features for disease categories. An autoencoder is an unsupervised feature construction technique that uses the structure of a neural network, trained to reproduce its own input as output, to learn a distributed representation of the original features. With one linear hidden layer, autoencoder is like the well-known principal component analysis, but better results can often be achieved using de-noising autoencoders that learn a robust representation from a noisy version of the input, or stacking multiple autoencoders to build a deep autoencoder.

We first expanded the feature space by computing higher-order combinations of the original features used in the baseline. For example, when a simple term frequency feature is considered together with assertion types, note types, note section types, and a moving window in the temporal dimension, it can be quickly expanded to many sparse features, such as "number of times hypertension is mentioned as positive in the assessment and plan section in a progress note, in the last three months", which may better capture the complexity of the data generating process. Having learned the features using the best of the autoencoder methods - 100 features with the stacked autoencoders - we developed two methods: one where there is a single model for all disease categories and a second method where there is a model for each of the disease-categories.

2.2 Deep Learning Combined with Modified Alternating Decision Tree

As an enhancement to the above approach, we developed a single model for all disease categories using a modified ADT, which first stratifies the training data into disease-specific sub-models and then uses a single loss function to

jointly learn all sub-models. By reducing the number of meta parameters needed to be determined empirically, this method not only simplified the learning process, but also reduced the risk of overfitting. Splitting the model by disease categories early on injects the prior belief that not all diseases are equal, and allows disease-specific features to be considered by each sub-model. As the contribution of a disease-specific feature on lowering training error depends on its path in the tree, an early stratification step makes sure the feature will not be rejected only because it is considered out of context. This creates a complexity-balanced tree based on the data complexity in each category, and only requires a single stop criterion.

3. Results

As seen in Table 1, the overall problem list accuracy, measured by the F_1 score, improves to 0.72 with the deep learning method using modified ADT from the baseline (F_1=0.70). The improvement relative to the baseline is statistically significant at $p<0.01$. The results for a straightforward application of deep learning without modified ADT are quite poor as their F_1 score is significantly lower than the baseline. Figure 1 shows performance improvement achieved by the deep learning with modified ADT method compared to the baseline on a disease-category basis. The Figure also shows frequency of problems for each category, and statistical significance (p-value) of the difference between two methods for each category. Statistically, the proposed method significantly improves performance for three high frequency categories.

Table 1. Problem List Generation Accuracy indicated by the F_1 Score.

Problem List Generation Method	F_1 Score
Baseline (manually engineered features + single ADT model)	0.70
Stacked Autoencoder features + single ADT model	0.52
Stacked Autoencoder features + per-category ADT models	0.68
Stacked Autoencoder features + modified single ADT model	0.72

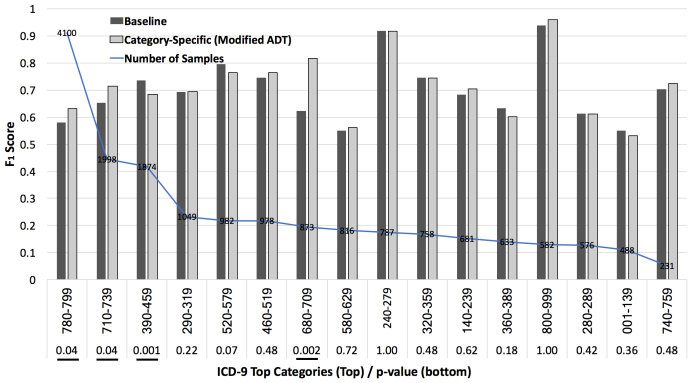

Figure 1. F_1 score comparison of the baseline and the autoencoder features + modified ADT method by ICD-9 top categories.

4. Discussion and Conclusion

Our experiments showed that deep learning helps a task like problem list generation in a couple of different ways. It has the potential to improve the overall accuracy, and even more importantly, it allows us to automatically learn features from data for multiple models. Straightforward unsupervised deep learning may not produce accurate models for relatively small amounts of data, which is often the case with clinical data. Our future work is aimed at investigating the methods with a larger amount of data. As is, however, autoencoder features already show promising results when combined with a modified ADT approach.

References

1. T. D. Shanafelt, L. N. Dyrbye, C. Sinskye, O. Hasan, D. Satele, J. Sloan and C. P. West, "Relationship Between Clerical Burden and Characteristics of the Electronic Environment With Physician Burnout and Professional Satisfaction," Mayo Clinic Proceedings, vol. 91, no. 7, pp. 836-848, 2016.
2. M. Devarakonda and C.-H. Tsou, "Automated Problem List Generation from Electronic Medical Records in IBM Watson," in Proceedings of the Twenty-Seventh Conference on Innovative Applications of Artificial Intelligence, Autin, TX, 2015.
3. Y. Bengio, "Learning deep architectures for AI," Foundations and Trends in Machine Learning, vol. 2, no. 1, pp. 1-127, 2009.
4. Department of Health and Human Services, "Medicare and Medicaid programs; electronic health record incentive program; final rule.," July 2010. http://edocket.access.gpo.gov/2010/pdf/2010-17207.pdf. [Accessed 8 March 2017].
5. Y. Freund and L. Mason, "The Alternating Decision Tree Algorithm," San Francisco, 1999.
6. Y. Bengio and A. Courville, "Representation learning: A review and new perspectives.," IEEE transactions on pattern analysis and machine intelligence, vol. 35, no. 8, pp. 1798-1828, 2013.
7. G. E. Hington and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks.," Science, vol. 313, no. 5786, pp. 504-507, 2006.